

# SPREAD-SPECTRUM TRANSCEIVER

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32

18  
a1

## Technical Field

This invention relates generally to digital communication systems, and more particularly to a spectrum spreading technique for use in multi-node digital communication systems such as digital networks and digital radios.

## Background of the Invention

Spectrum spreading techniques for use in digital communication networks have been described in many books and papers. A classic publication in this field is *Spread Spectrum Communications* by M. K. Simon, J. K. Omura, R. A. Scholtz and B. K. Levitt, Computer Science Press, 11 Taft Court, Rockville, Maryland 20850, 1985. Particular kinds of spectrum spreading techniques that have been implemented in digital communication networks in the prior art include "direct-sequence spreading", "frequency hopping", "time hopping", and various hybrid methods that involve combinations of the aforementioned techniques.

Multi-node spread-spectrum communication networks developed in the prior art were generally characterized as code-division multiple-access (CDMA) networks, which utilized "code-division multiplexing" (*i.e.*, a technique in which signals generated by different spreading-code sequences simultaneously occupy the same frequency band). Code-division multiplexing requires that the simultaneously used spreading codes be substantially "mutually orthogonal", so that a receiver with a filter matched to one of the spreading codes rejects signals that have been spread by any of the other spreading codes.

In a typical multi-node spread-spectrum communication network using either a conventional direct-sequence spectrum spreading technique, or a hybrid technique involving ~~e.g.~~ direct-sequence and frequency-hopped spectrum spreading, only a single spreading code is employed. At regular intervals, the polarity of the spreading code is either inverted (*i.e.*, each 0 is changed to 1, and each 1 is changed to 0) or left unchanged, depending

1 on whether the next bit of information to be transmitted is a 1 or a 0. The  
2 resulting signal is an "information-bearing" sequence, which ordinarily  
3 would be transmitted using some type of phase-shift keyed (PSK)  
4 modulation -- usually, binary phase-shift keyed (BPSK) modulation or  
5 quaternary phase-shift keyed (QPSK) modulation.

6 A publication entitled *Spread Spectrum Techniques Handbook*, Second  
7 Edition, March 1979, which was prepared for the National Security Agency  
8 by Radian Corporation of Austin, Texas, describes a number of  
9 spread-spectrum techniques that had been proposed in the prior art. Of  
10 particular interest is a direct-sequence technique described on  
11 page 2 - 21 *et seq.* of the *Spread Spectrum Techniques Handbook*, which  
12 involved transmitting one bit of information (either a 0 or a 1) by  
13 switching between two independent signals that are generated by  
14 different spreading codes. Ideally, the spreading codes of the two  
15 independent signals should be "almost orthogonal" with respect to each  
16 other, so that cross-correlation between the two sequences is very small.  
17 In practice, in such early spread-spectrum communication systems, the  
18 two independent signals were maximal-length linear recursive sequences  
19 (MLLRSs), often called "M-sequences", whose cross-correlations at all  
20 possible off-sets had been computed and found to be acceptably low.  
21 However, this technique of switching between two independent signals did  
22 not achieve widespread acceptance, mainly because it required  
23 approximately twice the electronic circuitry of a polarity-inversion  
24 technique without providing any better performance.

25 Two recent papers, viz., "Spread-Spectrum Multiple-Access  
26 Performance of Orthogonal Codes: Linear Receivers" by P. K. Enge and  
27 D. V. Sarwate, (*IEEE Transactions on Communications*, Vol. COM-35,  
28 No. 12, December 1987, pp. 1309 - 1319), and "Spread-Spectrum  
29 Multiple-Access Performance of Orthogonal Codes for Indoor Radio  
30 Communications" by K. Pahlavan and M. Chase, (*IEEE Transactions on*  
31 *Communications*, Vol. 38, No. 5, May 1990, pp. 574 - 577), discuss  
32 multi-node spread-spectrum communication networks in which multiple  
33 orthogonal sequences within a relatively narrow bandwidth are assigned  
34 to each node, whereby a corresponding multiplicity of information bits can

1 be simultaneously transmitted and/or received by each node -- thereby  
2 providing a correspondingly higher data rate. A specified segment of each  
3 sequence available to a node of the network is designated as a "symbol".  
4 In the case of a repetitive sequence, a symbol could be a complete period  
5 of the sequence. The time interval during which a node transmits or  
6 receives such a symbol is called a "symbol interval". In a multi-node  
7 spread-spectrum network employing multiple orthogonal sequences, all  
8 the nodes can simultaneously transmit and/or receive information-bearing  
9 symbols derived from some or all of the sequences available to the nodes.

10 The emphasis in the aforementioned Enge *et al.* and Pahlavan *et al.*  
11 papers is on network performance, especially in certain kinds of signal  
12 environments. Neither paper recommends or suggests using any particular  
13 set of mutually orthogonal spreading codes for generating multiple  
14 orthogonal sequences; and neither paper discloses how to derive or  
15 generate suitable mutually orthogonal spreading codes. However, methods  
16 of generating families of sequences that are pairwise "almost orthogonal"  
17 by using two-register sequence generators have been known for some  
18 time.

19 In a paper entitled "Optimal Binary Sequences for Spread-Spectrum  
20 Multiplexing" by R. Gold, (*IEEE Transactions on Information Theory*, Vol.  
21 IT-13, October 1967, pp. <sup>119</sup>619 - <sup>121</sup>621), so-called "Gold codes" were  
22 proposed for use as spreading codes in multi-node direct-sequence  
23 spread-spectrum communication networks of the CDMA type. A Gold code  
24 is a linear recursive sequence that is generated by a product  $f_1 f_2$ , where  $f_1$   
25 and  $f_2$  comprise the members of a so-called "preferred pair" of primitive  
26 polynomials of the same degree  $n$  over a field  $GF(2)$ . A primitive  
27 polynomial of degree  $n$  is defined as a polynomial that generates a  
28 maximal-length linear recursive sequence (MLLRS), which has a period of  
29  $(2^n - 1)$ . The required relationship between  $f_1$  and  $f_2$  that makes them a  
30 preferred pair is described in the aforementioned paper by R. Gold.

31 A Gold code is a particular kind of "composite code". Other kinds of  
32 composite codes include "symmetric codes" and "Kasami codes". A  
33 symmetric code is similar to a Gold code in being generated by a product

1  $f_1 f_2$  of a pair of primitive polynomials, except that for a symmetric code  
 2 the polynomial  $f_2$  is the "reverse" of primitive polynomial  $f_1$ , i.e.,  
 3  $f_2(x) = x^n f_1(1/x)$ , where  $n = \deg f_1 = \deg f_2$ . The correlation properties  
 4 of Gold codes and symmetric codes are discussed in a paper entitled  
 5 "Cross Correlation Properties of Pseudorandom and Related Sequences" by  
 6 ~~M. B. Pursley and D. V. Sarwate~~ <sup>D. V. Sarwate M. B. Pursley</sup>, (Proceedings of the IEEE, Vol. 68, <sup>No 5</sup> May  
 7 1980, pp. 593 - 619). Kasami codes differ from Gold codes in that for  
 8 Kasami codes, the polynomials  $f_1$  and  $f_2$  are not of the same degree.  
 9 Kasami codes are also discussed in the aforementioned paper by M. B.  
 10 Pursley and D. V. Sarwate. The concept of a "composite code" can be  
 11 broadened to include sequences obtained from a two-register sequence  
 12 generator, where the sequences generated in the two registers can be  
 13 quite general.

14 Predominant among the reasons that have militated against using  
 15 direct-sequence spreading codes for multi-node spread-spectrum  
 16 communication networks of the prior art is the so-called "near-far"  
 17 problem. If the nodes of a multi-node spread-spectrum communication  
 18 network are widely distributed so that power levels for different nodes  
 19 can differ markedly at a given receiver in the network, then at the given  
 20 receiver the correlations of a reference sequence with a sequence that is  
 21 transmitted by a nearby node are apt to be stronger than correlations of  
 22 the reference sequence with a version of the reference sequence that has  
 23 been transmitted from a greater distance. Adverse effects of the  
 24 "near-far" problem can include periodic strong correlations in  
 25 information-bit errors, and false synchronization. To avoid such adverse  
 26 effects, frequency hopping has been preferred in the prior art for  
 27 multi-node spread-spectrum communication networks -- especially for  
 28 tactical networks where the nodes are widely distributed. Until recently,  
 29 most of the research funding and efforts in connection with multi-node  
 30 spread-spectrum communication networks have been directed toward  
 31 tactical networks, thereby virtually precluding significant research on  
 32 direct-sequence spread-spectrum communication networks.

33 Hybrid frequency-hopped and direct-sequence spread-spectrum  
 34 communication networks have been proposed for tactical applications.

TOP SECRET

1 However, the frequency diversity provided by "hopping" of the carrier  
2 readily enables rejection of unintended signals, thereby making the choice  
3 of a particular spreading-code sequence relatively unimportant.  
4 Consequently, there has been substantially no research in the prior art on  
5 the use of Gold codes and other composite codes for hybrid  
6 frequency-hopped and direct-sequence spread-spectrum communication  
7 networks.

8 Direct-sequence spread-spectrum communication networks have  
9 received recent attention in connection with the development of wireless  
10 local area networks (LANs), personal communications networks (PCNs),  
11 and cellular telephone networks utilizing communications satellites. The  
12 "near-far" problem is ordinarily not an issue for LANs and PCNs, because  
13 the nodes in such networks are generally distributed at distances that are  
14 not very far from each other. For cellular telephones, the "near-far"  
15 problem is not an issue in satellite applications, because all transmitters  
16 in the "spot beam" from a satellite are roughly at the same distance from  
17 the satellite.

18 Several wireless LANs are described in an article entitled "Spread  
19 Spectrum Goes Commercial" by D. L. Schilling, R. L. Pickholtz and L. B.  
20 Milstein, *IEEE Spectrum*, <sup>Vol. 27, No. 8,</sup> August 1990, pp. 40 - 45. For indoor  
21 spread-spectrum communication networks (e.g., wireless LANs), spectrum  
22 spreading has commonly been employed in "star network" configurations.  
23 In a star network, the nodes are normally synchronized with a master  
24 controller, so that each node of the network can use a different offset of  
25 the same spreading-code sequence. False synchronization is not ordinarily  
26 encountered with star networks. In circumstances in which two or more  
27 star networks, each utilizing a different spreading-code sequence, operate  
28 in close proximity to each other, composite codes could be used to  
29 advantage to prevent interference between neighboring star networks.  
30 However, in the prior art, reliance has usually been placed upon the  
31 distance between the individual star networks, and upon  
32 signal-attenuating structures (e.g., walls) separating the individual star  
33 networks, as well as upon cross-correlation properties that are expected  
34 of random uncorrelated spreading-code sequences, to enable one star

1 network to reject signals from another star network in its vicinity.  
2 Consequently, composite codes have generally not been used in star  
3 networks.

4 In PCNs, the use of composite codes as spreading-code sequences  
5 has not yet received much attention, because factors such as size, weight  
6 and power considerations have generally favored simplicity over  
7 performance. Techniques involving satellite-based CDMA cellular radio  
8 networks have emerged from developments in wireless LANs, but have  
9 generally been concerned with coding and systems engineering rather than  
10 with spreading-code sequence generation.

11 To date, direct-sequence spectrum spreading techniques have been  
12 used primarily in applications requiring high multipath immunity, good  
13 time resolution, robustness, privacy and low probability of detection, and  
14 for which in-band interference and the "near/far" problem are  
15 manageable. Such applications have included satellite communications,  
16 star networks in office environments, mobile radio, and positioning and  
17 navigation applications. The use of composite codes (e.g., Gold codes or  
18 symmetric codes) for spectrum spreading in such applications has not  
19 heretofore been deemed appropriate, because composite codes would  
20 require significantly greater hardware complexity to implement than  
21 MLLRSs without seeming to provide sufficient compensating advantages  
22 over MLLRSs in terms of processing gain, the number of nodes that can be  
23 accommodated, the rate of data transmission, or robustness.

## 24 Summary of the Invention

25 It is a general object of the present invention to provide a  
26 spread-spectrum technique for use in a multi-node digital communication  
27 network, whereby a unique set of spreading-code sequences is assigned to  
28 each node of the network for transmitting digital signals.

29 It is a particular object of the present invention to provide a method  
30 for generating a family of nearly orthogonal spreading-code sequences,  
31 and for assigning a unique set of spreading-code sequences from the

1 family of sequences so generated to each node of a multi-node digital  
2 communication network.

3 It is also a particular object of the present invention to provide  
4 methods for selecting a set of one or more spreading-code sequences that  
5 can be used during a specified period of time (*i.e.*, a so-called "symbol  
6 interval") to convey multiple bits of information, if the selected sequence  
7 or sequences of the set are modulated and transmitted simultaneously.

8 It is likewise a particular object of the present invention to provide  
9 logic circuit designs for hardware implementation of methods for  
10 generating a family of spreading-code sequences for assignment to the  
11 nodes of a multi-node digital communication network.

12 It is a further object of the present invention to provide methods for  
13 simultaneously modulating a set of carriers of the same frequency but of  
14 different phases in order to enable multiple bits of information to be  
15 transmitted on each carrier of the set.

16 It is another object of the present invention to provide a  
17 spread-spectrum technique for use in a multi-node digital communication  
18 network, which can readily incorporate standard error-control coding  
19 (whose parameters are matched to the particular application) into the  
20 transmission and reception of digital signals propagated by the network.

21 It is also an object of the present invention to provide a technique  
22 whereby conventional equipment designed for generating arbitrary  
23 spreading-code sequences can be adapted to the task of generating a  
24 family of spreading-code sequences for use in a multi-node digital  
25 communication network.

26 It is a further object of the present invention to provide a technique  
27 whereby direct-sequence spectrum spreading, or a hybrid combination of  
28 direct-sequence and frequency-hopped spectrum spreading, can be utilized  
29 in conjunction with code diversity or "code hopping" in a spread-spectrum  
30 digital communication network designed to have a low probability of  
31 intercept (LPI).

1 It is also an object of the present invention to provide symbol  
2 detection methods, which enable a receiver at any given node in a  
3 multi-node spread-spectrum digital communication network to determine  
4 the most likely spreading-code sequence or sequences transmitted by  
5 another node of the network attempting to communicate with the given  
6 node.

## 7 Description of the Drawing

8 FIG. 1 is a schematic illustration of an apparatus for generating a  
9 family of nearly orthogonal spreading-code sequences of the composite  
10 code type, and for selecting unique sets of the sequences so generated for  
11 assignment to corresponding nodes of a multi-node digital communication  
12 network according to the present invention.

13 FIG. 2 is a schematic illustration of an alternative embodiment of a  
14 spreading-code sequence generator for use in the apparatus of FIG. 1,  
15 which allows register taps to be arbitrarily selected for summation (*i.e.*,  
16 "EXCLUSIVE OR") and feedback functions.

17 FIG. 3 is a schematic illustration of another alternative embodiment  
18 of a spreading-code sequence generator for use in the apparatus of FIG. 1,  
19 wherein one of the modulo-2 adders (*i.e.*, "EXCLUSIVE OR" circuits) shown in  
20 FIG. 1 is omitted, which enables a maximal-length linear recursive  
21 sequence (MLLRS) to be used as one of the possible spreading-code  
22 sequences.

23 FIG. 4 is a schematic illustration of yet another alternative  
24 embodiment of a spreading-code sequence generator for use in the  
25 apparatus of FIG. 1, which allows information to be transmitted by  
26 switching in register contents (called "fills") obtained from look-up  
27 tables at the beginning of each symbol interval.

28 FIG. 5 is a schematic representation of a procedure according to the  
29 present invention whereby two sequences are selected from the set of  
30 sequences that are available to a given node of the network for modulating  
31 two sinusoidal carriers, which are of the same frequency but which differ  
32 in phase by 90°.

FIG. 6 is a schematic representation of a procedure according to the present invention whereby the set of spreading-code sequences available to a given node of the network is partitioned into two subsets, and whereby sequences are selected from each of the subsets and modulated onto orthogonal carriers.

FIG. 7 is a schematic representation of a procedure according to the present invention whereby three sequences are selected from the set of sequences that are available to a given node of the network, and are combined so as to be capable in effect of modulating three sinusoidal carriers of the same frequency but with relative phases of  $0^\circ$ ,  $60^\circ$  and  $120^\circ$ .

FIG. 8 is a schematic representation of a procedure according to the present invention whereby four sequences are selected from the set of sequences that are available to a given node of the network, and are combined so as to be capable in effect of modulating four sinusoidal carriers of the same frequency but with relative phases of  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  and  $135^\circ$ .

FIG. 9 is a schematic representation of a procedure according to the present invention whereby externally generated spreading-code sequences serve as inputs to two shift registers for generating unique spreading-code sequences.

FIG. 10 is a block diagram of a transmitter for use by a node of a multi-node digital communication network according to the present invention.

FIG. 11 is a block diagram of a receiver for use by a node of a multi-node digital communication network according to the present invention.

FIG. 12 is a block diagram of a correlation unit of the receiver of FIG. 11, which correlates each in-coming spreading-code sequence detected by the receiver with all the spreading-code sequences that are available to the node.

## 1 Best Mode of Carrying Out the Invention

2 In accordance with the present invention, a family of "almost  
3 orthogonal" binary sequences is generated to provide disjoint sets of  
4 spreading-code sequences that can be assigned to corresponding nodes of a  
5 multi-node digital communication network. Each node of the network is  
6 allotted multiple spreading-code sequences, which are selected from the  
7 total number of available sequences provided by the family of "almost  
8 orthogonal" binary sequences. The spreading-code sequences assigned to  
9 the various nodes of the network are all modulo-2 sums (*i.e.*, "EXCLUSIVE OR"  
10 outputs) of the contents (also called the "fills") of successive stages in  
11 two so-called "shift registers".

12 The binary sequences from which the disjoint sets of spreading-code  
13 sequences are selected for assignment to the nodes of the network are  
14 said to be "almost orthogonal" because the selected binary sequences all  
15 have low auto-correlation values (except for offset 0), and all have low  
16 cross-correlation values relative to each other, where the  
17 auto-correlations and the cross-correlations are performed over a  
18 specified number of bits that defines a so-called "symbol interval". For  
19 algebraically generated periodic linear recursive sequences that are  
20 selected for their favorable auto-correlation and cross-correlation  
21 properties, the optimum symbol interval for a given sequence coincides  
22 with the period of the sequence. For sequences generated by a non-linear  
23 random number generator, and for linear recursive sequences of very long  
24 period, the symbol interval for a given sequence can be chosen  
25 arbitrarily -- in which case the auto-correlation and cross-correlation  
26 properties of the sequences cannot be guaranteed, but have the usual  
27 statistics for correlations of random sequences.

28 An example of a set of binary spreading-code sequences that could  
29 be used in a multi-node digital communication network according to the  
30 present invention would be a set of Gold code sequences, each of which is  
31 generated by the product  $f_1 f_2$  of a "preferred pair" ( $f_1, f_2$ ) of primitive  
32 polynomials of the same degree  $n$  over the field  $GF(2)$ , *i.e.*, the algebraic  
33 field of two elements 0 and 1. A primitive polynomial over  $GF(2)$  is a  
34 polynomial that generates a maximal-length linear recursive sequence

1 (MLLRS). If the degree of the primitive polynomials  $f_1$  and  $f_2$  is  $n$ , the  
 2 period of the Gold code sequences generated by the product  $f_1 f_2$  is  
 3  $(2^n - 1)$ .

4 Another example of a set of binary spreading-code sequences that is  
 5 suitable for use in a multi-node digital communication network would be a  
 6 set of so-called "symmetric" sequences, each of which is generated by  
 7 the product  $f_1 f_2$ , where  $f_1$  and  $f_2$  are primitive polynomials, and  
 8 where  $f_2$  is the "reverse" of  $f_1$ , i.e.,

$$9 \quad f_2(x) = x^n f_1(1/x),$$

10 where  $n = \deg f_1 = \deg f_2$ .

11 Yet another example of a set of binary spreading-code sequences  
 12 that could be employed in a multi-node digital communication network  
 13 according to the present invention would be a set of Kasami code  
 14 sequences, each of which is generated by a product  $f_1 f_2$ , where  $f_1$  and  $f_2$   
 15 are primitive polynomials such that the degree of one of the polynomials  
 16 divides the degree of the other.

17 The auto-correlation properties of composite-code sequences (e.g.,  
 18 Gold code sequences, symmetric code sequences and Kasami code  
 19 sequences), and the cross-correlation properties of families of such  
 20 composite-code sequences over an entire period, are described in the  
 21 aforementioned article by M. B. Pursley *et al.* wherein such sequences are  
 22 shown to be "almost orthogonal."

23 Alternatively, a set of random spreading-code sequences could also  
 24 be used in practicing the present invention. While composite-code  
 25 sequences are especially useful and convenient for particular  
 26 embodiments of a multi-node digital communication network according to  
 27 the present invention, it is not necessary to limit the invention in  
 28 principle to the use of any particular kinds of spreading-code sequences.  
 29 The salient characteristic of a network according to the present invention  
 30 is a two-register sequence generator, which enables multiple

1 spreading-code sequences to be obtained by combining the outputs of  
2 selected stages of each of the two registers.

3 Various embodiments of a multi-node digital communication  
4 network according to the present invention are described hereinafter. In  
5 each of these embodiments, a family of binary spreading-code sequences  
6 can be generated using Gold code sequences, or "symmetric" sequences, or  
7 Kasami code sequences, or any other suitable sequence generation scheme.  
8 From the family of binary spreading-code sequences so generated, a unique  
9 set of multiple spreading-code sequences is assigned to each node of the  
10 network. Specified codes, or their reciprocals (*i.e.*, codes of opposite  
11 polarity), are selected periodically for transmission by each node, where  
12 the particular codes and polarities that are selected in a particular case  
13 depend upon the information to be conveyed. Since information is  
14 conveyed in blocks, Reed-Solomon coding (or any other suitable coding  
15 scheme) can optionally be used to provide forward error control.

16 Symbol decision methods (*i.e.*, methods that can be used by a  
17 receiver to determine the most likely transmitted sequence or sequences)  
18 can vary for different embodiments of the present invention. In each  
19 embodiment, the receiver identifies those particular incoming sequences  
20 having the strongest correlation values, and determines their polarities.  
21 The decision logic algorithm for each embodiment determines the most  
22 likely transmitted sequence or sequences from the correlation values.

23 If Gold code sequences, or "symmetric" sequences, or Kasami code  
24 sequences are used as the spreading-code sequences, mathematically  
25 guaranteed cross-correlation properties of those sequences over an entire  
26 period can be exploited by taking the symbol interval to be equal to the  
27 period of the spreading-code sequences. According to one method for  
28 ensuring that modulation is "balanced" (*i.e.*, that equal numbers of 0's and  
29 1's are transmitted during each symbol interval), the symbol interval is  
30 taken to be equal to twice the period of the spreading-code sequences, and  
31 the spreading-code sequences are transmitted so that a complete  
32 sequence is transmitted during the first half of a symbol interval and so  
33 that the reciprocal of that sequence is transmitted during the second half  
34 of the symbol interval. This method produces a factor-of-two decrease in

1 the symbol rate for a given "chip rate" (*i.e.*, the rate at which individual  
2 bits of the spreading-code sequences are transmitted).

3 Acquisition and maintenance of synchronization for spread-spectrum  
4 signals have been widely discussed in published literature. In each  
5 embodiment of the present invention, synchronization of each incoming  
6 sequence with the spreading-code sequences that have been assigned to a  
7 given node is acquired by conventional means. Synchronization is  
8 maintained, and the possibility of false synchronization is minimized, by  
9 using a two-register sequence generator to generate candidate  
10 spreading-code sequences that are to be correlated with each incoming  
11 sequence. If synchronization of an incoming sequence with the sequences  
12 assigned to the given node is lost, that incoming sequence does not  
13 correlate strongly with any of the candidate spreading-code sequences.  
14 However, if synchronization is maintained, the incoming sequences that  
15 are most likely to be signals transmitted by other nodes of the network  
16 are determined. A stream of information bits is then assembled from the  
17 incoming sequences identified as likely to be information-bearing signals.  
18 If forward error correction has been used, the information bit stream is  
19 decoded to determine the information originating at the transmitting node  
20 of the network.

21 A specified number  $K$  of available spreading-code sequences is  
22 assigned to each node of a network according to the present invention.  
23 The number of information bits that can be conveyed per symbol varies  
24 directly with the value of the number  $K$ . If the total number of  
25 spreading-code sequences available to the network is  $N$ , then the  
26 maximum number of nodes that can be accommodated by the network is  
27  $N/K$ . Thus, there is a trade-off between the number of information bits  
28 that can be conveyed per symbol and the maximum number of nodes that  
29 can be accommodated by the network.

30 In embodiments of the present invention in which composite codes  
31 are employed, the individual spreading-code sequences assigned to a given  
32 node of the network may be specified by feedback taps associated with  
33 the polynomials  $f_1$  and  $f_2$ , and by the initial "fills" (*i.e.*, contents) of shift  
34 registers corresponding to the polynomials  $f_1$  and  $f_2$ . Various methods can

TD-20-540540

be used to specify the polynomials  $f_1$  and  $f_2$ , and to specify the initial fills of the  $f_1$ -register (i.e., the register whose feedback taps correspond to the polynomial  $f_1$ ) and the  $f_2$ -register (i.e., the register whose feedback taps correspond to the polynomial  $f_2$ ) for each node of the network. A preferred method is for the fill associated with the polynomial  $f_1$  to remain always the same for all the nodes of the network, and for the initial fill associated with the polynomial  $f_2$  for each particular node to be specified or derived from <sup>a specified</sup> fill. Thus, the unchanging fill for the  $f_1$ -register for every node of the network could consist of the so-called "impulse fill," i.e., a 1 as the content of the first stage of the register and 0's as the contents of the remaining stages of the register. If there are  $V$  nodes in the network and each node is identified by a corresponding integer  $v$ , where  $0 \leq v \leq V-1$ , and if  $K$  spreading-code sequences are assigned to each node, the initial fill for the  $f_2$ -register of the  $v$ th node could be obtained by first loading the  $f_2$ -register with the initial fill of the network controller (designated as "node 0"), and then stepping the  $f_2$ -register  $Kv$  times.

If composite codes are used for the spreading-code sequences, and if the number of "composite sequences" assigned to each node of the network equals or exceeds  $KV$ , where a "composite sequence" is the modulo-2 sum of a non-zero sequence generated by  $f_1$  and a non-zero sequence generated by  $f_2$ , the aforescribed method is sufficient for specifying the initial fills of the  $f_1$ -register and the  $f_2$ -register. For example, if Gold code sequences are used for which  $\deg f_1 = \deg f_2 = n$ , the aforescribed method is sufficient for specifying the initial fills of the  $f_1$ -register and the  $f_2$ -register, provided that  $KV \leq (2^n - 1)$ . If  $KV = 2^n$ , the MLLRS generated by either  $f_1$  or  $f_2$  must be used by one of the nodes as one of its symbols. If  $KV = (2^n + 1)$ , the MLLRS generated by  $f_1$  and the MLLRS generated by  $f_2$  must both be used (either both of the MLLRSs by one node, or each of the MLLRSs by a different node) as symbols. The assignment of initial fills to the two registers must then be modified accordingly.



are illustrated in FIG. 1 as being of the same size  $M$  (i.e., both have the same number of stages); although there is no requirement in principle that both of the shift registers 10 and 11 have the same number of stages. For the embodiment illustrated in FIG. 1, each of the shift registers 10 and 11 has a size indicated by the parameter  $M = 7$ , which indicates seven "stages" or "flip-flops". The number of stages provided in commercially available shift registers is usually a multiple of 8.

Each of the shift registers 10 and 11 is "driven" by a polynomial, which is one of a preferred pair of primitive polynomials  $f_1$  and  $f_2$  of degree  $n$ , where  $n \leq M$ . A set of feedback taps 12 is provided to "drive" the shift register 10, and a set of feedback taps 13 is provided to "drive" the shift register 11. For purposes of illustration, the polynomials  $f_1$  and  $f_2$  are of degree  $n = 5$ . The feedback taps 12 correspond to the polynomial

$$f_1(x) = 1 + x^2 + x^5;$$

and the feedback taps 13 correspond to the polynomial

$$f_2(x) = 1 + x + x^2 + x^4 + x^5.$$

A "symbol selection" unit 20 receives corresponding spreading-code sequences from the shift registers 10 and 11. The purpose of the symbol selection unit 20 is to select one or the other of the two spreading-code sequences produced by the shift registers 10 and 11 for transmission to a modulator during each specified symbol interval. The symbol selection unit 20 also receives a sequence of information bits provided by an information source 22. These information bits may be encrypted and encoded, as discussed hereinafter.

If  $2^r$  spreading-code sequences are available to each node of the network, the stream of information bits is partitioned into blocks of  $(r + 1)$  bits. The first  $r$  of these bits serve as an address in a table, which contains the number of <sup>the</sup> spreading-code sequences to be transmitted during the next symbol interval. The  $(r + 1)$ th bit is a "differential encoding" bit, which determines whether the sequence to be transmitted during the next symbol interval is "inverted" (i.e., complemented

TOP SECRET

1 modulo 2) or "upright" (*i.e.*, not inverted). Thus, if the  $(r + 1)$ th bit is  
2 a 1, the next transmitted sequence has a "polarity" opposite that of the  
3 current sequence; and if the  $(r + 1)$ th bit is a 0, the next transmitted  
4 sequence has the same "polarity" as the current sequence. For example, if  
5 the current sequence is upright, and the  $(r + 1)$ th bit is a 1, the next  
6 transmitted sequence is inverted. Similarly, if the current sequence is  
7 upright, and the  $(r + 1)$ th bit is a 0, the next transmitted sequence is  
8 upright.

9 The technique of partitioning information bits into blocks of bits  
10 (*i.e.*, the "blocking" of encrypted bits) as described above is especially  
11 well suited to the use of Reed-Solomon spreading-code sequences. In the  
12 foregoing example in which  $2^r$  spreading-code sequences are available to  
13 each node of the network,  $(r + 1)$ -bit blocks of information are  
14 interpreted by a Reed-Solomon encoder as elements of the finite field  
15  $GF(2^r + 1)$ . These field elements are assembled into blocks to which  
16 redundant field elements are appended in accordance with the particular  
17 Reed-Solomon coding scheme used. A discussion of Reed-Solomon codes is  
18 found in a text by F. J. MacWilliams and N. J. A. Sloane entitled *The Theory*  
19 *of Error Correcting Codes*, North-Holland Publishing Company, New York,  
20 (1978), pp. 301 - 305. Reed-Solomon codewords are then furnished to the  
21 symbol selection unit 20, which uses each field element of  $(r + 1)$ -bits  
22 to select a sequence and a polarity for transmission during the next  
23 symbol interval.

24 In FIG. 2, a more general configuration for the spreading-code  
25 sequence generator is shown, which enables the individual register taps to  
26 be arbitrarily selected for the summation (*i.e.*, EXCLUSIVE OR) and feedback  
27 functions. In the configuration of FIG. 2, the locations of the feedback  
28 taps are not "hardwired", but are programmable. Thus, the particular  
29 generating polynomials  $f_1$  and  $f_2$  can be arbitrarily assigned, and can be  
30 changed periodically if desired. As indicated in FIG. 2, parameters  
31  $t_0, \dots, t_6$  and  $s_0, \dots, s_6$  represent corresponding stages in the shift  
32 registers 10 and 11, respectively. Each of the parameters  $t_0, \dots, t_6$  and  
33  $s_0, \dots, s_6$  takes the value 1 or 0 according as the corresponding register  
34 stage is tapped or not tapped.

1        Regardless of the type of sequence generator used (*i.e.*, whether of  
 2        the "hardwired" type as illustrated in FIG. 1 or of the programmable type  
 3        as illustrated in FIG. 2), if the sequence of 0's and 1's emanating from a  
 4        particular stage of one register (*e.g.*, the "bottom stage" of the upper  
 5        register as shown in either FIG. 1 or FIG. 2) is denoted by  $\{a_k\}$ , and if the  
 6        sequence of 0's and 1's emanating from a correspondingly particular stage  
 7        of the other register (*e.g.*, the "top stage" of the lower register as shown  
 8        in FIG. 1 or FIG. 2) is denoted by  $\{b_k\}$ , the  $(2M - 1)$  spreading-code  
 9        sequences available from the modulo-2 adders are

$$10 \quad \{a_k + b_{k-i}\}, \text{ where } i = 1, 2, \dots, M - 1, \text{ and}$$

$$11 \quad \{a_{k-i} + b_k\}, \text{ where } i = 0, 1, \dots, M - 1.$$

12        These spreading-code sequences,  $\{a_k + b_{k-i}\}$  and  $\{a_{k-i} + b_k\}$ , are distinct  
 13        from each other. In the case where Gold code sequences are used, the  
 14        sequences  $\{a_k + b_{k-i}\}$  and  $\{a_{k-i} + b_k\}$  constitute a subset of size  $(2M - 1)$   
 15        of a set of  $(2^n + 1)$  non-zero linear recursive sequences generated by the  
 16        polynomial product  $f_1 f_2$ . Only  $(2^n - 1)$  of the  $(2^n + 1)$  spreading-code  
 17        sequences generated by the polynomial product  $f_1 f_2$  have the product  $f_1 f_2$   
 18        as their "minimal polynomial". The other two sequences, *viz.*,  $\{a_k\}$  and  
 19         $\{b_k\}$ , are generated individually by polynomials  $f_2$  and  $f_1$ , respectively.

20        The sequences  $\{a_k\}$  and  $\{b_k\}$  may be accessed by omitting one of the  
 21        adders shown in FIG. 1, thereby obtaining sequences generated by  $f_1$  or  $f_2$   
 22        alone, as illustrated in FIG. 3.

23        When  $M < 2^r \leq 2M - 1$ , it is advantageous for the  $2^r$  spreading-code  
 24        sequences that are available to each node of the network to be allocated  
 25        between a subset of  $2^{r-1}$  so-called "upper sequences" of the form  
 26         $\{a_k \oplus b_{k-i}\}$  and a subset of  $2^{r-1}$  so-called "lower sequences" of the form  
 27         $\{a_k \oplus b_{k-i}\}$ . However, when  $2^r \leq M$ , it is preferable for all of the  
 28        spreading-code sequences to be selected from either the upper sequences  
 29        or the lower sequences. Within a given subset (*e.g.*, a subset consisting  
 30        only of the upper sequences, or a subset consisting only of the lower

1 sequences), the cross-correlations between different spreading-code  
 2 sequences are effectively correlations between different offsets of the  
 3 same maximal-length linear recursive sequence (MLLRS) and have the  
 4 value  $-1$ , which is very small compared to the length of the sequence  
 5  $(2^n - 1)$ . In contrast, the correlation between a sequence selected from  
 6 the subset of upper sequences and a sequence selected from the subset of  
 7 lower sequences has a magnitude of either 1 or  $2^{[(n+1)/2]}$ , assuming Gold  
 8 code sequences are used, where  $2^{[(n+1)/2]}$  is small compared to  $(2^n - 1)$   
 9 but large compared to 1. Thus, if  $2^r \leq M$ , optimal cross-correlation  
 10 properties among all the spreading-code sequences assigned to a given  
 11 node can be assured by selecting all of the spreading-code sequences from  
 12 the same subset of either upper sequences or lower sequences. If  
 13  $M < 2^r \leq 2M - 1$ , optimal cross-correlation properties among all the  
 14 spreading-code sequences assigned to a given node can be substantially  
 15 achieved by selecting  $2^{r-1}$  spreading-code sequences from each of the  
 16 subsets of upper and lower sequences, and by using an appropriate symbol  
 17 detection scheme as described hereinafter.

18 When the two correlations of largest magnitude from among all the  
 19 correlations between each of the candidate spreading-code sequences  
 20 assigned to a particular node and an incoming spreading-code sequence  
 21 received by that node are so close in magnitude that it is impossible on  
 22 the basis of the correlation values alone to determine reliably which one  
 23 of those two candidate sequences is the "correct" sequence (*i.e.*, the  
 24 sequence bearing the information intended for that particular node), the  
 25 following procedure can then be initiated to determine the "correct"  
 26 sequence. The set of  $2^r$  spreading-code sequences is considered to consist  
 27 of two subsets, *viz.*, the "upper sequences" and the "lower sequences"  
 28 described above, each of which consists of  $2^{r-1}$  sequences. For each of  
 29 the two subsets, a "punctured" sum of the correlation magnitudes (*i.e.*, the  
 30 sum of all the correlation values except the largest one) is computed. The  
 31 subset having the smaller "punctured" sum is then assumed to be the  
 32 "correct" subset, *i.e.*, to contain the "correct" spreading-code sequence.  
 33 The "correct" spreading-code sequence is then identified as the sequence

1 within the "correct" subset that has the largest correlation magnitude  
2 with respect to the incoming spreading-code sequence.

3 The rationale for assuming that the "correct" spreading-code  
4 sequence (i.e., the sequence bearing the information intended for the  
5 particular node) is contained in the subset having the smaller "punctured"  
6 sum is grounded on the fact that the correlation values between different  
7 sequences within the "correct" subset must all have a magnitude of 1,  
8 whereas the magnitudes of the correlation values of spreading-code  
9 sequences in different subsets are either 1 or  $2^{[(n+1)/2]}$  with equal  
10 probability. Consequently, when an errorless spreading-code sequence is  
11 correlated with all of the  $2^r$  spreading-code sequences that are candidates  
12 for selection, the "punctured" sum of the correlation magnitudes for the  
13 subset containing the "correct" incoming sequence is  $(2^{r-1} - 1)$ , whereas  
14 the "punctured" sum of the correlation magnitudes that would be expected  
15 for the subset containing an "incorrect" incoming sequence is

$$2^{r-2} + (2^{r-2} - 1) 2^{[(n+1)/2]},$$

17 assuming that the correlation magnitudes for spreading-code sequences  
18 from the "incorrect" subset are divided equally between the values 1 and  
19  $2^{[(n+1)/2]}$ . The ratio between the largest and the smallest "punctured"  
20 sums, which may be considered as the "expected margin" between the  
21 subset containing the "correct" sequence and the subset containing an  
22 "incorrect" sequence, is approximately  $2^{[(n-1)/2]}$ .

23 The foregoing analysis assumes that  $2^{r-2}$  of the  $2^{r-1}$  sequences in  
24 the "incorrect" subset have correlation magnitudes of 1 with respect to  
25 the "correct" incoming sequence, and that the  $2^{r-2}$  remaining sequences  
26 in the "incorrect" subset have correlation magnitudes of  $2^{[(n+1)/2]}$ .  
27 However, this assumption actually only represents an average condition.  
28 As  $r$  increases in value within the range  $2^r \leq 2M - 1$ , the assumption  
29 becomes more accurate, provided that each of the correlation magnitudes  
30 1 and  $2^{[(n+1)/2]}$  independently occurs with a probability of 0.5. This  
31 "balance" between the subsets of upper and lower sequences increases as

1 the value of  $r$  increases. Thus, the probability of selecting the "correct"  
 2 subset increases as the number  $2^r$  of spreading-code sequences increases.

3 The "symbol decision" logic by which the spreading-code sequences  
 4 assigned to the individual nodes of a multi-node digital communications  
 5 network as illustrated in FIG. 1 are selected is described as follows. Let  
 6  $L$  and  $N$  denote the spreading-code sequences corresponding to the largest  
 7 and the next-largest correlation magnitudes, respectively, of a set of  $2^r$   
 8 "symbols" (*i.e.*, sequences). For purposes of this discussion, the  
 9 designations  $L$  and  $N$  can denote both the sequences and also the  
 10 magnitudes of the correlations of these sequences with the received  
 11 signal. To determine the "correct" symbol, first compute the ratio  
 12  $R = L/N$ , and then compare  $R$  with a selectable first threshold value  $T_1$ . If  
 13  $R > T_1$ , choose  $L$ . If  $R \leq T_1$ , then a "symbol decision" algorithm is utilized  
 14 as follows:

- 15 1) If  $L$  and  $N$  are sequences in the same subset, declare an  
 16 erasure. If  $L$  and  $N$  are not in the same subset, then for each of  
 17 the two subsets compute the sum of all correlation magnitudes  
 18 except the largest correlation magnitude in each subset  
 19 (*i.e.*, except  $L$  and  $N$ ). Denote the subset corresponding to the  
 20 smaller of these two sums by  $S_1$ , and the subset corresponding  
 21 to the larger of these two sums by  $S_2$ . Let  $N_1$  denote the  
 22 next-largest correlation magnitude in  $S_1$ .
- 23 2) If  $L$  is in  $S_1$  and  $N$  is in  $S_2$ , compare the ratio  $L/N_1$  with a  
 24 selectable second threshold value  $T_2$ . If  $L/N_1 > T_2$ , choose  $L$ . If  
 25  $L/N_1 \leq T_2$ , then declare an erasure.
- 26 3) If  $L$  is in  $S_2$  and  $N$  is in  $S_1$ , then if  $N/N_1 > T_2$ , choose  $N$ ; and if  
 27  $N/N_1 \leq T_2$ , declare an erasure.

28 Using the foregoing algorithm, it is possible for strong correlations  
 29 between candidate spreading-code sequences and the information-bearing  
 30 sequences that are actually transmitted by other nodes of the network to  
 31 be rejected. Regardless of whether all the candidate spreading-code  
 32 sequences are selected from the same subset of upper or lower sequences,

1 or are equally divided between sequences from each subset, a "symbol  
2 decision" error can occur when a signal from an unintended node of the  
3 network strongly correlates with one of the candidate spreading-code  
4 sequences, or when a sequence belonging to the intended node correlates  
5 more strongly than does the "correct" sequence with the received signal.  
6 The probability of such a strong correlation occurring decreases as the  
7 number  $2^r$  of spreading-code sequences per node increases. Thus, the use  
8 of multiple spreading-code sequences per node not only provides  
9 robustness, but also reduces the effect of the "near-far" problem.

10 In principle, any number of temporally contiguous bits can be  
11 designated as a "symbol". However, if composite code sequences (e.g.,  
12 Gold code sequences, symmetric sequences, or Kasami code sequences) are  
13 used as the spreading-code sequences, advantageous auto-correlation and  
14 cross-correlation properties can be guaranteed only if the correlations  
15 are performed over an entire period of each sequence in the family of  
16 possible sequences. Thus, it is advantageous to designate the entire  
17 period of a composite code sequence as the "symbol". If each node of the  
18 network can use  $2^r$  spreading-code sequences, then each symbol  
19 represents  $r$  bits. The "inverse" (or "reciprocal") of a symbol is formed  
20 by replacing each 0 by a 1, and each 1 by a 0. By transmitting the inverse  
21 of a symbol along with the symbol, an additional bit of differentially  
22 encoded information per symbol can be transmitted. Thus, the information  
23 rate that can be achieved using a network as illustrated in FIG. 1 is

$$\frac{c(r + 1)}{(2^n - 1)}$$

24 where  $c$  is the number of chips (i.e., bits of the spreading-code sequence)  
25 transmitted per second.

26 To <sup>ensure</sup> insure that there is a balance between the number of 1's and 0's  
27 transmitted, a symbol interval could be taken to be equal to the duration  
28 of two periods of a spreading-code sequence. Opposite polarities of the  
29 spreading-code sequence would be transmitted during the first and second  
30 halves of the symbol interval. This technique would increase the  
31 processing gain, but would decrease the information rate by a factor of 2.  
32

In practice, it should not be necessary to use this technique if the information-bearing sequence is random, because polarity inversions of random sequences occur approximately half the time anyway. Input sequence randomizers are commonly employed in digital communication systems, and use of such an expedient can be assumed where appropriate in practicing the present invention.

In an alternative embodiment of the present invention as illustrated in FIG. 4, only one spreading-code sequence is selected for transmission during a given symbol interval. After a particular symbol has been transmitted, appropriate register fills for the next symbol are "looked up" from a table and are "switched in." Where the registers are driven by polynomials (as where composite codes are used for the spreading-code sequences), the use of a "look up" table is a preferred embodiment that minimizes hardware requirements for the transmitter (but not for the receiver). In FIG. 4, the last two stages of each of the registers 10 and 11 are unnecessary, because the number of bits in the "switched-in" fills need be no greater than the degrees of the polynomials that generate the linear recursive sequences. Furthermore, in the embodiment of FIG. 4, the number of spreading-code sequences that can be assigned to each node is not limited by the register length  $M$  as is the case in the embodiment of FIG. 1 in which the number of sequences available to the node is bounded above by  $2M - 1$ .

The technique described above for transmitting information by using multiple "almost-orthogonal" spreading-code sequences according to the present invention provides performance advantages over other modulation schemes that have been used in the prior art. According to the technique described above, the number of bits of information per symbol increases as the number  $2^r$  of spreading-code sequences increases, yet the "distance" between symbols (i.e., the cross-correlation values of the sequences) does not change as the number  $2^r$  of spreading-code sequences increases. This is contrary to the usual situation encountered in digital communication systems that utilize, e.g., quadrature-amplitude modulation (QAM).

In QAM systems, amplitude-phase states function as symbols. Thus, an increase in the number of amplitude-phase states results in an increase in the information rate of a QAM system, but also results in an increase in the bit-error rate. The increase in the bit-error rate occurs because, for a given average energy level, the amplitude-phase states become "closer" to each other in the Euclidean sense (*i.e.*, the distance between adjacent amplitude-phase states decreases) as the number of amplitude-phase states increases, thereby making them harder to distinguish from each other. However, where orthogonal spreading-code sequences function as symbols, as in the present invention, the symbols are never "close" to each other regardless of the number of symbols used. Consequently, for systems that utilize orthogonal spreading-code sequences, the symbol error rate does not increase as rapidly as the information rate when the number of symbols increases.

In TABLE I, values for various performance-measuring parameters are listed as functions of the parameters  $n$  and  $K$  for a network according to a first embodiment of the present invention as illustrated in FIGS. 1 - 4. A "chip rate" (*i.e.*, the number of bits of the spreading-code sequence transmitted per second) of 2.5 MHz is arbitrarily assumed, although in practice the chip rate can be chosen to optimize system parameters such as bandwidth and information rate for the particular application. If a different chip rate were to be used, the information rate could be obtained by multiplying the appropriate value in the last column of TABLE I (*i.e.*, the BPSK modulation rate) by  $c/2.5$  MHz, where  $c$  is the number of chips transmitted per second expressed in MHz. The embodiment of FIGS. 1 - 4 is operated in a mode in which a single spreading-code sequence modulates a carrier to generate a BPSK signal, where  $n$  is the degree of the polynomials  $f_1$  and  $f_2$  used for generating the spreading-code sequences, and where  $K$  is the number of sequences per user.

Also listed in TABLE I are useful measures of processing gain for different degrees of the polynomials  $f_1$  and  $f_2$ . The first number in each entry in the column labelled "Processing Gain" is the value for  $10 \log_{10} (2^n - 1)$  expressed in dB, which represents the processing gain

1 against other spreading-code sequences assigned to the given node,  
 2 assuming that synchronization is maintained and that the correct subset  
 3 is chosen (when applicable, as discussed above). The second number,  
 4 which is shown in parentheses, in each entry in the column labelled  
 5 "Processing Gain" represents the processing gain against spreading-code  
 6 sequences transmitted by other nodes in the network, using the same  
 7 polynomials  $f_1$  and  $f_2$  for generating the spreading-code sequences.

8 **TABLE I**

9

Degree $n$	Processing Gain (dB)	Sequences per Node $K = 2^r$	Information Rate (bits/period) $r + 1$	Number of Nodes $2^{n-r}$	Information Rate (kbits/sec) BPSK (2.5 MHz)
8	24 (12)	16	5	16	49.0
8		32	6	8	58.8
9	27 (13)	16	5	32	24.5
9		32	6	16	29.4
10	30 (15)	32	6	32	14.7
10		64	7	16	17.1
11	33 (16)	32	6	64	7.3
11		64	7	32	8.5
12	36 (13)	16	5	256	3.1
12		32	6	128	3.7
12		64	7	64	4.3
13	39 (19)	32	6	256	1.8
14	42 (21)	32	6	512	0.9
14		64	7	256	1.1

10

11 **Embodiment II:**

12 In an alternative embodiment of the present invention, as illustrated  
 13 in FIG. 5, two spreading-code sequences are selected from among all the  
 14 available spreading-code sequences generated by the shift registers 10  
 15 and 11 during each period of the sequences. The selected sequences are  
 16 used to modulate the "in-phase" arm and/or the "quadrature" arm, (also  
 17 called the I-arm and the Q-arm), respectively, of a sinusoidal carrier.

1 Modulation of the I-arm and the Q-arm can be achieved using a quaternary  
 2 phase-shift keyed (QPSK) modulation, an offset QPSK (also called an  
 3 OQPSK) modulation, a quadrature partial response (QPR) modulation, or any  
 4 other type of quadrature modulation. If  $K$  spreading-code sequences are  
 5 available to each node of the network, there are

$$6 \quad \frac{K(K-1)}{2}$$

7 possible pairs of spreading-code sequences that can be transmitted per  
 8 symbol interval. Thus, by selecting two of the  $K$  available spreading-code  
 9 sequences for transmission during each symbol interval,

$$10 \quad \left[ \log_2 \frac{K(K-1)}{2} \right]$$

11 bits of information can be conveyed per symbol.

12 If the polarities of the spreading-code sequences can be selectively  
 13 inverted or not inverted, another information bit can be conveyed per  
 14 symbol so as to increase the total number of bits of information that can  
 15 be conveyed per symbol to

$$16 \quad 1 + \left[ \log_2 \frac{K(K-1)}{2} \right].$$

17 Thus, for example, if  $K = 9$ , the number of information bits per symbol is  
 18  $1 + [\log_2 36] = 6$ . The two sequences to be transmitted during each  
 19 symbol interval are chosen by table lookup. Whether or not to invert the  
 20 spreading-code sequences is determined by differential encoding of one of  
 21 the six bits.

22 In TABLE II, values for various performance-measuring parameters  
 23 are listed as functions of the parameters  $n$  and  $M$  for a network as  
 24 illustrated in FIG. 5, again assuming a chip rate of 2.5 MHz. The  
 25 spreading-code sequence generator shown in FIG. 4 has a coherent  
 26 receiver, so as to be able to distinguish and track the I-arm and the Q-arm  
 27 of the carrier. It is possible that a given spreading-code sequence could

- 1 appear in the I-arm during one symbol interval, and in the Q-arm during
- 2 another symbol interval.

TABLE II

Degree $n$	Processing Gain (dB)	Sequences per Node $K$	Information Rate (bits/period) $\frac{1 + [\log_2 K(K-1)]}{2}$	Number of Nodes $[(\frac{2^n + 1}{K})]$	Information Rate (kbits/sec) $\frac{2^n K}{2.5 \text{ MHz}}$
8	24 (12)	9	6	28	68.6
8		12	7	21	78.4
9	27 (13)	9	6	57	34.2
9		12	7	42	39.1
9		17	8	30	44.0
9		24	9	21	48.9
10	30 (15)	9	6	113	17.1
10		12	7	85	19.6
10		17	8	60	22.0
10		24	9	42	24.4
11	33 (16)	9	6	227	8.5
11		12	7	170	9.8
11		17	8	120	11.0
11		24	9	85	12.2
11		33	10	62	13.4
12	36 (18)	9	6	455	4.3
12		12	7	341	4.9
12		17	8	241	5.5
12		24	9	170	6.1
12		33	10	124	6.7

### Embodiment III:

In a third embodiment of the present invention as illustrated in FIG. 6, two spreading-code sequences are selected during each symbol interval, viz., one "upper" sequence and one "lower" sequence from each of the shift registers 10 and 11. If the number of spreading-code sequences available to each node of the network is  $K = 2^r$ , each subset contains  $2^{r-1}$

1 sequences, so that  $2(r - 1)$  bits of information can be transmitted per  
2 symbol interval. If the polarity of each spreading-code sequence is  
3 selectively inverted, or not, according to a differential coding scheme,  
4 then  $2 + [2(r - 1)] = 2r$  information bits per symbol interval are  
5 transmitted. For example, if  $K = 8$ , then six information bits per symbol  
6 are transmitted.

7 Since symbol decisions are made within each subset of  
8 spreading-code sequences, there is no need to choose the "correct" subset  
9 in order to identify the "correct" spreading-code sequence. Thus, decision  
10 logic is considerably simplified. Also, symbol decisions are made  
11 between sequences that have optimal cross-correlation properties.

12 In TABLE III, values are given for the same performance parameters  
13 as listed above for the first and second embodiments, again assuming a  
14 chip rate of 2.5 MHz.

TOP SECRET

TABLE III

Degree <i>n</i>	Processing Gain (dB)	Sequences per Node	Information Rate (bits/period)	Number of Nodes	Information Rate (kbits/sec) -BPSK (2.5 MHz)
8	24 (12)	8	6	32	58.8
8		16	8	16	78.4
8		32	10	8	98.0
9	27 (13)	8	6	64	29.3
9		16	8	32	39.1
9		32	10	16	48.9
9		64	12	8	58.7
10	30 (15)	8	6	128	14.7
10		16	8	64	19.6
10		32	10	32	24.4
10		64	6	16	29.3
11	33 (16)	8	6	256	7.3
11		16	8	128	9.8
11		32	10	64	12.2
11		64	12	32	14.7
12	36 (18)	8	6	455	4.3
12		16	8	256	4.9
12		32	10	128	6.1
12		64	12	64	7.3

**Embodiment IV:**

In a fourth embodiment of the present invention as illustrated in FIG. 7, three spreading-code sequences are selected during each symbol interval for simultaneous transmission using phase-shift keyed (PSK) modulation. The sequence generators shown in FIG. 7 are substantially the same as shown in FIG. 5, except that three spreading-code sequences (rather than two as shown in FIG. 5) are selected and transmitted to the modulator. The three spreading-code sequences are used to modulate a carrier having three components, which are 60° out of phase.

1 Besides the processing gain available due to the quasi-orthogonality  
2 of the spreading-code sequences in the embodiment illustrated in FIG. 7,  
3 the phase difference between carriers provides an additional 6 dB of  
4 processing gain, as can be seen by computing the correlation between two  
5 sinusoidal signals that are 60° out of phase.

6 If the number of spreading-code sequences available to the node is  
7  $K$ , the number of information bits that can be transmitted per symbol  
8 interval (including one bit corresponding to whether the spreading-code  
9 sequences are transmitted "upright" or "inverted") is given by

10 
$$1 + \left[ \log_2 \frac{K(K-1)(K-2)}{6} \right].$$

11 In TABLE IV, values are given for the same performance  
12 parameters as listed above for the first, second and third embodiments,  
13 again assuming a chip rate of 2.5 MHz.

FOR SECRET

TABLE IV

Degree $n$	Processing Gain (dB)	Sequences per Node	Information Rate (bits/period)	Number of Nodes	Information Rate (kbits/sec) <del>BPSK</del> (2.5 MHz)
8	12	9	7	28	68.6
8		11	8	23	78.4
8		14	9	18	88.2
8		17	10	15	96.0
8		20	11	12	105.6
9	13.5	9	7	57	34.2
9		11	8	46	39.1
9		14	9	36	44.0
9		17	10	30	48.9
9		20	11	25	53.8
10	15	9	7	113	17.1
10		11	8	93	19.5
10		14	9	73	22.0
10		17	10	60	24.4
10		20	11	51	26.8
11	16.5	9	7	227	8.5
11		11	8	186	9.8
11		14	9	146	11.1
11		17	10	120	12.3
11		20	11	102	13.5
12	18	9	7	455	4.3
12		11	8	372	4.9
12		14	9	292	5.5
12		17	10	241	6.1
12		20	11	204	6.7

3

4 *Embodiment V:*

5 In FIG. 8, a fifth embodiment of the present invention is illustrated,  
 6 in which four spreading-code sequences are transmitted per symbol  
 7 interval using "quaternion" phase-shift keyed modulation. The sequence

1 generators shown in FIG. 8 are substantially the same as shown in FIG. 5,  
2 except that four spreading-code sequences (rather than two as shown in  
3 FIG. 5) are selected and transmitted to the modulator. The four  
4 spreading-code sequences are used to modulate a carrier having four  
5 components, which are 45° out of phase.

6 Besides the processing gain available due to the quasi-orthogonality  
7 of the spreading-code sequences in the embodiment illustrated in FIG. 8,  
8 the phase difference between carriers provides an additional 3 dB of  
9 processing gain, as can be seen by computing the correlation between two  
10 sinusoidal signals that are 45° out of phase.

11 If the number of spreading-code sequences available to a node is  $K$ ,  
12 the number of information bits that can be transmitted per symbol  
13 interval (including one bit corresponding to whether the spreading-code  
14 sequences are transmitted "upright" or "inverted") is given by

15 
$$1 + \left[ \log_2 \frac{K(K-1)(K-2)(K-3)}{24} \right].$$

16 For example, if  $K = 8$ , the number of information bits that can be  
17 transmitted per symbol is 7. In TABLE V, values are given for the same  
18 performance parameters as listed above for the other embodiments, again  
19 assuming a chip rate of 2.5 MHz.

TABLE V

Degree $n$	Processing Gain (dB)	Sequences per Node	Information Rate (bits/period)	Number of Nodes	Information Rate (kbits/sec) BPSK (2.5 MHz)
8	12	8	7	32	68.6
8		10	8	25	78.4
8		11	9	23	88.2
8		13	10	19	98.0
8		15	11	17	107.8
8		17	12	15	117.6
9	13.5	8	7	64	34.2
9		10	8	51	39.1
9		11	9	46	44.0
9		13	10	39	48.9
9		15	11	34	53.8
9		17	12	30	58.7
10	15	8	7	128	17.1
10		10	8	102	19.6
10		11	9	93	22.0
10		13	10	78	24.4
10		15	11	68	26.9
10		17	12	60	29.3
11	16.5	8	7	256	8.5
11		10	8	204	9.8
11		11	9	186	11.0
11		13	10	157	12.2
11		15	11	136	13.4
11		17	12	120	14.7
12	18	8	7	512	4.3
12		10	8	409	4.9
12		11	9	372	5.5
12		13	10	315	6.1
12		15	11	273	6.7
12		17	12	241	7.3

1    **Embodiment VI:**

2        The foregoing embodiments I, II, III, IV and V of the present  
3    invention can be used for multi-node digital communication networks  
4    operating in modes in which spreading-code sequences are the sums of  
5    linear recursive sequences generated using feedback taps in each register  
6    of a two-register sequence generator. However, for privacy purposes, a  
7    multi-node digital communication network according to the present  
8    invention could also be used in a "code hopping" mode in which the  
9    spreading-code sequences are derived from externally generated  
10   sequences. Use of a communication network according to the present  
11   invention in a "code hopping" mode illustrates the power of the  
12   two-register configuration in preventing false synchronization, and in  
13   providing multiple information bits per symbol regardless of the manner  
14   of generating the spreading code.

15        A "code hopping" technique according to the present invention is  
16   illustrated in FIG. 9, which indicates switching at regular intervals  
17   between different spreading-code sequences, where each "input" sequence  
18   is arbitrarily selected and may be externally generated by a sequence  
19   generator 23. The switching intervals can be independent of any  
20   periodicities associated with input sequences. One or more input  
21   sequences may be selectively transmitted during a given switching  
22   interval, just as in the other embodiments. The particular input sequence  
23   or sequences selected for transmission during a given switching interval  
24   are determined by the symbol selection unit 20 on the basis of the  
25   information bits to be conveyed (as in the above-described embodiments),  
26   or on the basis of "cipher bits" used to maximize privacy by code hopping.  
27   In the code hopping mode, information is conveyed by polarity inversions,  
28   just as in ordinary direct-sequence spread-spectrum communications. In  
29   general, there is no necessary relationship between the information rate  
30   and the code hopping rate.

31        The previous embodiments I, II, III and IV can be used for either  
32   synchronous operation (*i.e.*, all nodes of the network are synchronized to a  
33   central node) or asynchronous operation (*i.e.*, synchrony is obtained only  
34   when communication takes place). In the "code hopping" embodiment,

1 however, synchronous operation is necessary because the externally  
2 generated spreading-code sequences are unique to each node, and  
3 communication between nodes must be coordinated by a central controller.

4 In a code hopping mode, low cross-correlation between  
5 spreading-code sequences is not guaranteed. In fact, the  
6 cross-correlation statistics for spreading-code sequences in a "code  
7 hopping" mode are similar to the cross-correlation statistics for random  
8 sequences. For example, if the symbol interval contains 2047 chips,  
9 approximately 5% of the correlation values should exceed  $\sqrt{2047} \approx 90$ . By  
10 contrast, if a Gold Code is used, the maximum correlation magnitude is  
11 only  $1 + 2^6 = 65$ . Thus, symbol errors are considerably more likely to  
12 occur in a "code hopping" mode than in a mode in which composite codes  
13 are used for the spreading-code sequences, and in which switching  
14 between spreading-code sequences occurs at intervals equal to the period  
15 of the sequences. However, a "code-hopping" technique could be effective,  
16 provided error-correction coding is used. It is noteworthy that in some  
17 star-networked local area networks, the correlation statistics of random  
18 sequences are accommodated with acceptable bit error rates.

19 A transmitter for each node of a multi-node digital communication  
20 network according to the present invention is illustrated schematically in  
21 FIG. 10 in which the spreading-code sequence generator of FIG. 1 is  
22 indicated by the reference number 30. Output from the sequence  
23 generator 30 serves as input for a modulator 31, which can use a  
24 conventional modulation technique such as BPSK, QPSK, OQPSK, etc. As  
25 also shown in FIG. 10, output from an information source 32 is encrypted  
26 by an encryption unit 33, which could optionally use the Data Encryption  
27 Standard certified by the National Bureau of Standards.

28 Encrypted output from the encryption unit 33 serves as input to a  
29 Reed-Solomon encoder 34, which is programmable to specify information  
30 rates that are appropriate for the specified embodiment, and for the  
31 particular mode of operation (e.g., using Gold code sequences, random  
32 sequences, etc.). Error-control coded output from the Reed-Solomon

1 encoder 34 serves as input to a symbol selection unit 35, which could be  
2 implemented in software on a commercially available microprocessor.

3 The symbol selection unit 35 selects one or more candidate  
4 spreading-code sequences from among all the spreading-code sequences  
5 available to a particular node of the network for input to the modulator  
6 31. The modulator 31 modulates the outputs of the sequence generator  
7 30 onto a carrier for transmission. A signal encoded in accordance with  
8 the present invention is then transmitted by the modulator 31 to the  
9 various nodes of the network.

10 A receiver for each node of a network according to the present  
11 invention is illustrated schematically in FIG. 11 in which the  
12 spreading-code sequence generator of FIG. 1 is indicated by the reference  
13 number 30. A synchronization-and-tracking unit 36 is used to maintain  
14 continuous communications. Synchronization and tracking techniques for  
15 spread-spectrum systems are well-developed in the art, and form the  
16 subject of an expansive body of literature. A demodulator 37 heterodynes  
17 the spread-spectrum signal to baseband. In the case of a hybrid  
18 frequency-hopped direct-sequence implementation, the demodulator 37  
19 provides baseband chip-synchronized data to a symbol recovery unit 38,  
20 which makes symbol decisions and provides the bits associated with each  
21 recovered symbol to a Reed-Solomon decoder 39.

22 As shown in FIG. 12, the symbol recovery unit 38 of FIG. 11 includes  
23 a correlation unit 41 and a symbol detection and logic unit 42. The  
24 symbol recovery unit 38 correlates the input signal with each candidate  
25 spreading-code sequence. The symbol detection and logic unit 42  
26 determines the strongest correlation outputs, makes a decision on the  
27 most likely transmitted sequence or sequences, and makes  
28 symbols-to-bits assignments. The Reed-Solomon decoder 39 of FIG. 11  
29 processes the recovered symbols, and passes the decoded bitstream to a  
30 decryptor 40, if encryption is to be used.

31 The present invention has been described above in terms of  
32 particular classes of spreading-code sequences, a particular type of  
33 error-control coding (viz., Reed-Solomon coding), constrained numbers of